Amendments to the Claims:

This listing of claims will replace all prior versions, and listings, of claims in the application:

Please amend the claims as follows:

1. (Currently Amended) A method comprising:

defining a severity level for a first controller, the first controller processing messages based on the defined ~~effective~~ severity level, wherein, available severity levels include:

a) a first severity level that indicates the existence of an anomaly that an application can recover from, said application also able to perform a desired task;

b) a second severity level that indicates the existence of an error that an application can recover from, said application also being unable to perform a desired task;

wherein, said determining includes selecting one of a), b) above;

defining a minimum severity level and a maximum severity level with a second controller, the second controller being a child controller to the first controller in a controller hierarchy so as to receive said severity level as a parameter from said first controller; and

setting the severity level of the second controller equal to the severity level of the first controller if the severity level of the first controller is between the minimum severity level and the maximum severity level, said first and second controllers [[each]] being:

~~an instance of a~~ tracing controller instances; or,

~~an instance of a~~ logging controller instances.

2. (Original) The method as in claim 1 further comprising:

setting the severity level of the second controller equal to the minimum severity level if the severity level of the first controller is relatively lower than the defined minimum severity level.

3. (Original) The method as in claim 2 further comprising:

setting the severity level of the second controller equal to the maximum severity level if the severity level of the first controller is relatively higher than the defined maximum severity level.

4. (Previously Presented) The method as in claim 1, wherein said first and second controllers are tracing controllers, further comprising:

receiving a trace message directed at the second controller;

comparing a severity value associated with the trace message with the severity level of the second controller; and

processing the trace message if the severity value associated with the trace message is greater than or equal to the severity level of the second controller.

5. (Original) The method as in claim 4 wherein processing comprises formatting information contained in the trace message and forwarding the formatted information to a specified output destination.

6. (Original) The method as in claim 4 wherein the trace message is emitted in response to the execution of a specified area of program code within an application.

7. (Previously Presented) The method as in claim 1, wherein said first and second controllers are logging controllers, further comprising:

receiving a log message directed at the second controller;

comparing a severity value associated with the log message with the severity level of the second controller; and

processing the log message if the severity value associated with the trace message is greater than or equal to the severity level of the second controller.

8. (Previously Presented) The method as in claim 1, wherein said first and second controllers are logging controllers, and wherein processing comprises formatting information contained in a log message and forwarding the formatted information to a specified output destination.

9. (Original) The method as in claim 7 wherein the second controller is associated with a particular category and wherein the log message is from an application and/or network component associated with the category.

10. (Currently Amended) A method comprising:

defining an severity level for a first controller, the first controller processing messages based on the defined severity level, wherein, available severity levels include:

a) a first severity level that indicates the existence of an anomaly that an application can recover from, said application also able to perform a desired task;

b) a second severity level that indicates the existence of an error that an application can recover from, said application also being unable to perform a desired task;

wherein, said determining includes selecting one of a), b) above;

defining a minimum severity level and a maximum severity level with a second controller, the second controller being a child controller to the first controller in a defined controller hierarchy~~ so as to receive said severity level as a parameter from said first controller~~; and

setting the severity of the second controller equal to the minimum severity level if the severity level of the first controller is relatively lower than the defined minimum severity level and setting the severity of the second controller equal to the maximum severity level if the severity level of the first controller is relatively higher than the defined maximum severity level, said first and second controllers [[each]] being:

[[an]] instances [[of]] a tracing controllers; or,

[[an]] instances [[of]] a logging controllers.

11. (Original) The method as in claim 10 further comprising:

setting the severity of the second controller equal to the severity of the first controller if the severity level of the first controller is between the minimum severity level and the maximum severity level.

12. (Original) The method as in claim 11 further comprising:

setting the severity of the second controller equal to the maximum severity level if the severity level of the first controller is relatively higher than the defined maximum severity level.

13. (Previously Presented) The method as in claim 10, wherein said first and second controllers are tracing controllers, further comprising:

receiving a trace message directed at the second controller;

comparing a severity value associated with the trace message with the severity of the second controller; and

processing the trace message if the severity associated with the trace message is greater than or equal to the severity of the second controller.

14. (Original) The method as in claim 13 wherein processing comprises formatting information contained in the trace message and forwarding the formatted information to a specified output destination.

15. (Original) The method as in claim 13 wherein the trace message is emitted in response to the execution of a specified area of program code within an application.

16. (Previously Presented) The method as in claim 10, wherein said first and second controllers are logging controllers, further comprising:

receiving a log message directed at the second controller;

comparing a severity value associated with the log message with the severity of the second controller; and processing the log message if the severity associated with the trace message is greater than or equal to the severity of the second controller.

17. (Original) The method as in claim 10 wherein processing comprises formatting information contained in a log message and forwarding the formatted information to a specified output destination.

18. (Original) The method as in claim 16 wherein the second controller is associated with a particular category and wherein the log message is from an application and/or network component associated with the category.

19. (Currently Amended) A system comprising:

a first controller to process messages based on a defined severity level, wherein, available severity levels include::

a) a first severity level that provides information for debugging;

b) a second severity level that indicates when a method is entered and/or exited;

c) a third severity level that echoes what has been performed; and

a second controller being a child controller to the first controller in a defined controller hierarchy ~~so as to receive said severity level as a parameter from said first controller~~; and

severity generation logic to set the severity of the second controller equal to the severity of the first controller if the severity level of the first controller is between a minimum severity level and a maximum severity level, said first and second controllers [[each being:

[[an]] instances of [[a]] tracing controllers[[; or,]]

~~an instance of a logging controller~~.

20. (Original) The system as in claim 19 wherein the severity generation logic sets the severity of the second controller equal to the minimum severity level if the severity level of the first controller is relatively lower than the defined minimum severity level.

21. (Original) The system as in claim 20 wherein the severity generation logic sets the severity of the second controller equal to the maximum severity level if the severity level of the first controller is relatively higher than the defined maximum severity level

22. (Previously Presented) The system as in claim 19, wherein said first and second controllers are tracing controllers, further comprising:

severity comparison logic to compare a severity value associated with a trace message directed at the second controller with the severity of the second controller, the second controller to process the trace message only if the severity associated with the trace message is greater than or equal to the severity of the second controller.

23. (Original) The system as in claim 22 further comprising:

a formatting module to format information contained in the trace message prior to forwarding the formatted information to a specified output destination

24. (Original) The system as in claim 22 wherein the trace message is emitted in response to the execution of a specified area of program code within an application.

25. (Previously Presented) The system as in claim 19, wherein said first and second controllers are logging controllers, further comprising:

severity comparison logic to compare a severity value associated with a log message directed at the second controller with the severity of the second controller, the second controller to process the log message only if the severity associated with the log message is greater than or equal to the severity of the second controller.

26. (Previously Presented) The system as in claim 19, wherein said first and second controllers are tracing controllers, further comprising:

a formatting module to format information contained in a trace message prior to forwarding the formatted information to a specified output destination.

27. (Original) The system as in claim 26 wherein the second controller is associated with a particular category and wherein [[the]] a log message is from an application and/or network component associated with the category.

28. (Currently Amended) An article of manufacture having program code stored thereon which, when executed by a machine cause the machine to perform the operations of:

defining a severity level for a first controller, the first controller processing messages based on the defined effective severity level, wherein, available severity levels include:

a) a first severity level that provides information for debugging;

b) a second severity level that indicates when a method is entered and/or exited;

c) a third severity level that echoes what has been performed;

defining a minimum severity level and a maximum severity level with a second controller, the second controller being a child controller to the first controller in a controller hierarchy so as to receive said severity level as a parameter from said first controller; and

setting the severity level of the second controller equal to the severity level of the first controller if the severity level of the first controller is between the minimum severity level and the maximum severity level, said first and second controllers [[each]] being:

[[an]] instances of [[a]] tracing controllers[[; or,]]

~~an instance of a logging controller~~.

29. (Original) The article of manufacture as in claim 28 comprising additional instructions to cause the machine to perform the operations of:

setting the severity level of the second controller equal to the minimum severity level if the severity level of the first controller is relatively lower than the defined minimum severity level.

30. (Original) The method as in claim 29 comprising additional instructions to cause the machine to perform the operations of:

setting the severity level of the second controller equal to the maximum severity level if the severity level of the first controller is relatively higher than the defined maximum severity level.